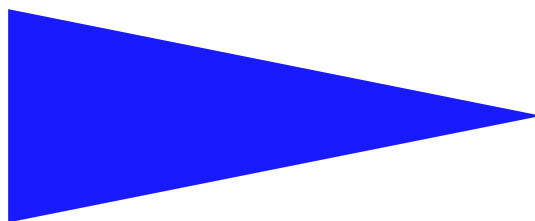


IRISA  
INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTÈMES ALÉATOIRES

PUBLICATION  
INTERNE  
N° 898



VISUAL SERVOING IN ROBOTICS SCHEME  
USING A CAMERA/LASER-STRIPES SENSOR

DJAMEL KHADRAOUI , GUY MOTYL , PHILIPPE MARTINET ,  
JEAN GALLICE , FRANÇOIS CHAUMETTE



CAMPUS UNIVERSITAIRE DE BEAULIEU - 35042 RENNES CEDEX - FRANCE



# Visual Servoing in Robotics Scheme Using a Camera/Laser-Stripe Sensor

Djamel Khadraoui \*, Guy Motyl \*\*, Philippe Martinet \*\*\*,  
Jean Gallice \*\*\*\*, François Chaumette \*\*\*\*\*

Programme 4 — Robotique, image et vision  
Projet Temis

Publication interne n° 898 — janvier 1995 — 26 pages

**Abstract:** The work presented in this paper fits into the realm of robotics and computer vision. The problem we seek to solve is the accomplishment of robotics tasks by using visual information provided by a particular sensor, mounted on a robot end effector. This sensor consists of two laser stripes fixed rigidly to a camera, projecting light planar on the scene.

First, we explain the approaches employed in robot control, using information provided by a vision sensor. More specifically, we explain the *visual servoing* approach, which corresponds to the one we have developed. Then, we express the modelling of visual information from a scene consisting of spherical objects, by using several representations in the image. This modelling permits us to establish a relation between the variations of visual information and camera velocities. Finally, both in simulation and in our experimental cell, results are presented. They concern the positioning task with respect to a sphere, and show the robustness and the stability of the control scheme.

**Key-words:** Vision-based control, laser stripe, task function, spherical objects

*Submitted to IEEE Transactions on Robotics and Automation*

(Résumé : *tsvp*)

\*LASMEA, Université Blaise Pascal, 63177 Aubière cedex, e-mail khadra@alize.univ-bpclermont.fr

\*\*TSI, IUT de St-Etienne, 42023 St-Etienne cedex, e-mail motyl@univ-stetienne.fr

\*\*\*LASMEA, Université Blaise Pascal, 63177 Aubière cedex, e-mail martinet@alize.univ-bpclermont.fr

\*\*\*\*LASMEA, Université Blaise Pascal, 63177 Aubière cedex, e-mail gallice@alize.univ-bpclermont.fr

\*\*\*\*\*IRISA / INRIA Rennes, e-mail chaumett@irisa.fr

# Asservissement visuel en robotique à partir d'un capteur caméra/laser

**Résumé :** Le travail présenté dans cet article s'intègre dans le thème de la robotique et de la vision par ordinateur. Le problème que nous cherchons à résoudre est l'accomplissement de tâches robotiques à partir d'informations visuelles fournies par un capteur particulier, monté sur l'effecteur d'un robot. Ce capteur est constitué de deux faisceaux laser couplés rigidement à une caméra, projetant des plans de lumière sur la scène. En premier lieu, nous décrivons les approches utilisées dans la commande des robots à partir d'informations fournies par un capteur de vision, notamment l'approche *asservissement visuel* qui correspond à celle que nous avons utilisée. Ensuite, nous abordons la modélisation des informations visuelles pour une scène constituée d'objets sphériques, en utilisant plusieurs représentations dans l'image. Cette modélisation nous permet d'établir une relation entre les variations d'une information visuelle et les mouvements de la caméra. Enfin, nous présentons les résultats obtenus, soit en simulation, soit sur notre site expérimental. Ils concernent la tâche de positionnement par rapport à une sphère, et montrent la robustesse et la stabilité de la commande.

**Mots-clé :** asservissement visuel, couplage caméra/laser, fonction de tâche, objets sphériques

# 1 Introduction

Nowadays, the vision sensor (camera) is more and more essential in order to resolve complex problems of the environment perception. Its miniaturization and recent image processing developments have allowed, first, the mounting of the visual sensor on the end effector of a robot, and secondly, the integration of visual information in a robot control loop.

These developments have permitted the realization of many more robotics tasks such as target tracking and obstacle avoidance. The first work concerning the use of visual information in robot control was conducted by Sanderson and Weiss [17], [21]. It presents two separate approaches. The first approach, commonly called “*Look and Move*”, is synthesized in terms of regulation of the end effector situation [2], [3], [20]. Thus, in this approach, an interpretation step of the end effector situation is necessary. This step is usually obtained with some inaccuracies, depending on the visual sensor geometry, environment and robot models. Moreover, the search for the end effector situation is time consuming and may affect the system’s overall behaviour.

The second approach, which removes the drawbacks of the previous approach, controls the end effector of a robot using visual information. This control scheme corresponds to the one we developed and is called “*visual servoing*”.

This paper is organized as follows: Section 2 describes the visual servoing approach and gives an overview of different aspects employed in robot control using information provided by a vision sensor. In the case of the camera-laser sensor, we express in Section 3 the modelling of the visual information obtained from a sphere, by using several representations in the image. We present results obtained in simulations and in our experimental cell in Section 4, in order to prove the robustness and the stability of the control scheme.

## 2 Visual Servoing Approach

### 2.1 Task Function

In this approach, the control is directly specified in terms of regulation in the image. One can notice that this approach has the advantage of avoiding the intermediary step of the estimation of the 3D situation  $\underline{r}$  of the workpiece with respect to the end effector.

For a given robotics task, a *target image* is built, corresponding to the desired position of the end effector with regard to the workpiece (see Figure 1). Then, a control scheme is developed directly based on image errors between the current image and the target image [4], [7].

It may be shown that all visual servoing tasks can be expressed as a regulation to zero of a function  $\underline{e}_1(\underline{r}, t)$ , called *vision-based task function* [16]. So, the use of a vision sensor allows us to build up such a task function which will be used in visual servoing. It is expressed by the following relation [16]:

$$\underline{e}_1(\underline{r}, t) = C[\underline{s}(\underline{r}, t) - \underline{s}^*] \quad (1)$$

where

- $\underline{s}^*$  is considered as a reference target image to be reached in the image frame.
- $\underline{s}(\underline{r}, t)$  is the value of visual features currently observed by the camera. These features depend on the situation between the sensor and the scene (noted  $\underline{r}$ ).

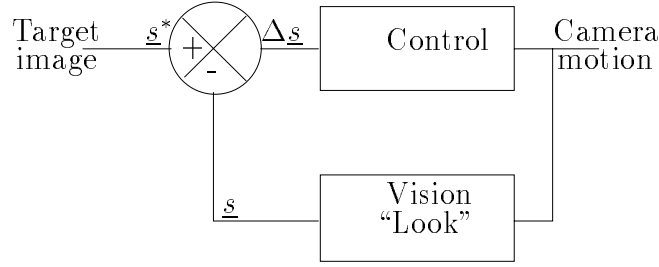


Figure 1: Visual servoing

- $C$  is a constant matrix, which takes into account more visual features than the number of robot degrees of freedom.

For a given task, the problem consists of choosing relevant visual features to achieve the task, and then constructing the constant matrix  $C$ . This matrix requires the establishment of the interaction matrix related to the chosen visual features. The choice of matrix  $C$  will be discussed later.

## 2.2 Interaction Matrix

We formalize the problem in terms of *sensor-based-control* [6] applied to visual servoing. The visual data, provided by the wrist-mounted sensory apparatus, is modelled as a set of *elementary signals*  $\underline{s}$  associated to the 2D geometric primitives in the image corresponding to the projection of the 3D primitives in the scene. The interaction between the sensor and the scene is described by a *coupling matrix*  $L_{\underline{s}}^T$  which links the behaviour of the signal to the sensor and/or object motion:

$$\dot{\underline{s}} = L_{\underline{s}}^T \cdot \xi \quad (2)$$

where

- $\dot{\underline{s}}$  is the time variation of  $\underline{s}$ .
- $\xi$  is the object velocity with respect to the sensor, with  $\xi = (\underline{T}, \underline{\Omega}) = (T_x, T_y, T_z, \Omega_x, \Omega_y, \Omega_z)^T$ .
- $L_{\underline{s}}^T$ , called *interaction matrix*, links the variations of  $\underline{s}$  and camera velocities. It perfectly expresses the interaction between the robot and its environment.

The first work concerning the modelling of visual information was developed by Feddema [9], [10]; it concerns the modelling of points and permits to control four degrees of freedom of a robot (three translations and one rotation).

Next, Espiau et al. modelled a set of low level geometrical primitives such as points, lines, circles, cylinders and spheres in the case of passive vision [4], [7]. Our work consists in the modelling of visual information by using a sensor called “*active*” in the sense that it is constituted by a camera and two laser stripes.

## 2.3 Redundancy Tasks

Each vision-based task can be conveyed as the regulation to zero of a function  $\underline{e}_1(\underline{r}, t)$ , defined by (1) which depends on the manipulator robot position and on the object motion [16]. This motion is parametrizable by the time variable  $t$ .

Now, we define the construction of a global task function  $\underline{e}(\underline{r}, t)$  given  $\underline{e}_1(\underline{r}, t)$ . Indeed, when only some degrees of freedom are used in a vision-based task, it may be interesting to use the other degrees of freedom in a secondary task. Generally, the realization of a secondary goal is expressed as a minimization of a cost function  $h_s(\underline{r}, t)$  under the constraint that the main task is achieved, i.e  $\underline{e}_1(\underline{r}, t) = 0$ .

If we consider a main task  $\underline{e}_1(\underline{r}, t)$  of dimension  $m$  and a secondary task  $g_s^T = \frac{\partial h_s}{\partial \underline{r}}$ , the determination of degrees of freedom which are left free by the main task requires the computation of the kernel of the matrix  $J_1 = \frac{\partial \underline{e}_1}{\partial \underline{r}}$ . In practice, this is equivalent to the knowledge of a  $m \times n$  full rank matrix  $W$  (where  $n$  is the number of robot degrees of freedom) such that :

$$Ker(W) = Ker(J_1) \quad (3)$$

Once such matrix is known, a possible global task function may be found [15], [16]:

$$\underline{e} = W^+ \underline{e}_1 + (\mathbf{I}_n - W^+ W) g_s^T \quad (4)$$

where  $W^+$  is the pseudo-inverse of  $W$  and  $(\mathbf{I}_n - W^+ W)$  an orthogonal projection operator which projects  $g_s^T$  in the kernel of  $J_1$ .

In the visual servoing case, the global task function can be expressed by:

$$\underline{e} = W^+ C [\underline{s}(\underline{r}, t) - \underline{s}^*] + (\mathbf{I}_n - W^+ W) g_s^T \quad (5)$$

Since the interaction matrix is defined by  $L_s^T = \frac{\partial \underline{s}}{\partial \underline{r}}$ , we can write:

$$J_1 = \frac{\partial \underline{e}_1}{\partial \underline{r}} = C L_s^T \quad (6)$$

and we can easily show from (3) that  $W$  can be chosen such that:

$$Ker(W) = Ker(L_s^T) \quad (7)$$

provided  $C$  satisfies  $CL^T$  of full rank.

## 2.4 Control Law and Stability

A basic control law consists of trying to obtain the task error  $\underline{e}(\underline{r}, t)$  so that it approximately behaves like a first-order decoupled system [7]. Therefore, we can express:

$$\dot{\underline{e}} = -\lambda \underline{e} \quad (8)$$

with  $\lambda > 0$ . Then, knowing that  $\underline{e}(\underline{r}, t)$  depends both on the manipulator robot position and the object motion, we can write:

$$\dot{\underline{e}} = \frac{\partial \underline{e}}{\partial \underline{r}} \dot{\underline{r}} + \frac{\partial \underline{e}}{\partial t} \quad (9)$$

where  $\xi_c$  is the camera velocity. Hence,  $\xi_c$  should ideally be of the form:

$$\xi_c = -\left(\frac{\partial \underline{e}}{\partial \underline{r}}\right)^{-1} (\lambda \underline{e} + \frac{\partial \underline{e}}{\partial t}) \quad (10)$$

where  $\left(\frac{\partial \underline{e}}{\partial \underline{r}}\right)$  and  $\frac{\partial \underline{e}}{\partial t}$  have to be determined.

It can be shown [7] that a sufficient condition for exponential convergence is given by:

$$CL_{\underline{s}}^T W^T > 0 \quad (11)$$

This relation allows us to choose the control matrix  $C$  from  $W$  and  $L_{\underline{s}}^T$ . For that, we use the pseudo inverse of the interaction matrix related to  $\underline{s}^*$ . This choice corresponds to:

$$C = WL_{\underline{s}=\underline{s}^*}^{T+} \quad (12)$$

where  $L_{\underline{s}=\underline{s}^*}^{T+}$  is the value of the interaction matrix at a location corresponding to the desired feature  $\underline{s} = \underline{s}^*$ . The positivity condition (11) is then satisfied in the neighbourhood of  $\underline{s} = \underline{s}^*$  since the product  $WW^T$  is positive. Moreover, it can be shown that this choice of  $C$  allows us to consider in (10)  $\frac{\partial \underline{e}}{\partial \underline{r}} = \mathbf{I}_n$ .

Furthermore, since  $W$  is constant, and by using the visual task function (4), we have:

$$\frac{\partial \underline{e}}{\partial t} = W^+ \frac{\partial \underline{e}_1}{\partial t} + (\mathbf{I}_n - W^+ W) \frac{\partial g_s^T}{\partial t} \quad (13)$$

Vector  $\frac{\partial \underline{e}_1}{\partial t}$  represents the contribution of a possible autonomous target motion and is generally unknown [7]. Consequently, if the object is motionless ( $\frac{\partial \underline{e}_1}{\partial t} = 0$ ), the control law (10) can finally be written:

$$\begin{cases} \xi_c = -\lambda \underline{e} - (\mathbf{I}_n - W^+ W) \frac{\partial g_s^T}{\partial t} \\ \text{with } \underline{e} = W^+ W L_{\underline{s}=\underline{s}^*}^T [\underline{s}(\underline{r}, t) - \underline{s}^*] + (\mathbf{I}_n - W^+ W) g_s^T \end{cases} \quad (14)$$

This simple control law only requires the tuning of gain  $\lambda$ , which depends on the rate of the control law and the robot dynamic (in our case  $\lambda = 0.1$ ).

## 2.5 Coupling a Camera and Laser Stripe

The use of a laser stripes allows us to reduce illumination problems. A camera alone detects more information about the image than necessary, and therefore the computing time of the image processing task is high because of the complexity of the scene. Laser stripes removes this drawback because only the information given by the laser stripes projection onto the scene is detected by the visual sensor.

Laser stripes in robotics have been widely used in real-time tracking of moving objects [19], and also in many applications allowing the recognition and interpretation of a workpiece surface [1]. The particular aim of these applications was to search for the three dimensional information of the visualized objects. In our applications, we use two laser stripes rigidly attached to the



camera, which is fixed to a robot manipulator. Each stripe projects a light planar onto the scene, which is static. We do not search for the three dimensional information and we mainly use visual information provided by the laser stripe projection onto the scene.

Moreover, the observed visual features are very straightforward to detect and depends only on the object geometry. The features are limited to points of discontinuity or lines in a polyhedral scene case [11], [12], [18]. Therefore, image processing is significantly reduced, hence a saving of time which furthers the dynamics of the system.

The only constraint imposed by the laser stripe is to know the position of the laser plane with respect to the camera frame. This can be obtained using classical calibration technique [11]. Knowing the laser plane parameters and also the geometry of the scene objects, we can model visual information from the image. After the modelling of interaction matrices, we can build the control scheme given by (14) allowing to accomplish a visual servoing.

We now present a general method for the modelling of these interaction matrices. Let us consider an elementary visual signal  $\underline{s}$  provided by this sensor. This camera-laser feature characterizes usual geometrical properties of the scene in the image. It is defined as the function  $\underline{s} = f(\underline{p}(\underline{r}))$  which depends on the configuration of the considered primitives, represented by the parameters  $\underline{p}$ . These parameters  $\underline{p}$  depend on the position  $\underline{r}$  between the sensor and the primitive. So, the time variation of  $\underline{s}$  can be obtained as:

$$\dot{\underline{s}} = \frac{\partial \underline{s}}{\partial \underline{p}} \frac{\partial \underline{p}}{\partial \underline{r}} \dot{\underline{r}} \quad (15)$$

where

- $\dot{\underline{s}}$  represents the time variation of  $\underline{s}$  in the image,
- $\dot{\underline{r}}$  is nothing but  $\xi$ , the object velocity with respect to the sensor ( $\xi = \dot{\underline{r}} = \frac{d\underline{r}}{dt}$ ).

We then have the interaction matrix  $L_{\underline{s}}^T$  expressed by:

$$L_{\underline{s}}^T = \frac{\partial \underline{s}}{\partial \underline{r}} = \frac{\partial \underline{s}}{\partial \underline{p}} \frac{\partial \underline{p}}{\partial \underline{r}} \quad (16)$$

The computation of  $\frac{\partial \underline{s}}{\partial \underline{p}}$  is generally trivial and we will see in the next section how to compute  $\frac{\partial \underline{p}}{\partial \underline{r}}$ .

- **Remark:** If we choose a new representation of the considered primitive, parametered by the function  $\underline{q} = \underline{q}(\underline{p})$ , which depends on the initial parameters  $\underline{p}$ , we can express  $\dot{\underline{q}}$  as follows:

$$\dot{\underline{q}} = \frac{\partial \underline{q}}{\partial \underline{p}} \dot{\underline{p}} \quad (17)$$

and then the corresponding interaction matrix is given by:

$$L_{\underline{s}}^T = \frac{\partial \underline{s}}{\partial \underline{q}} \frac{\partial \underline{q}}{\partial \underline{p}} \frac{\partial \underline{p}}{\partial \underline{r}} \quad (18)$$

In the past [18], this method has been used to compute interaction matrices related to polyhedral scenes. In that case, visual features only consist of points of discontinuity or lines. But, this method is more general since it can be applied for any geometrical primitive. In the next section, we present the case of a spherical scene by using several representations in the image.

### 3 Modelling Visual Features Obtained from a Sphere

In order to model visual features obtained from a spherical scene, it is necessary to select the ones which can be used in the control scheme. Then, we have to compute the related interaction matrix.

The sphere (see Figure 2) is represented by its centre  $\underline{m}_0 = (x_0 \ y_0 \ z_0)^T$  and its radius  $r$ , i.e:

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 - r^2 = 0 \quad (19)$$

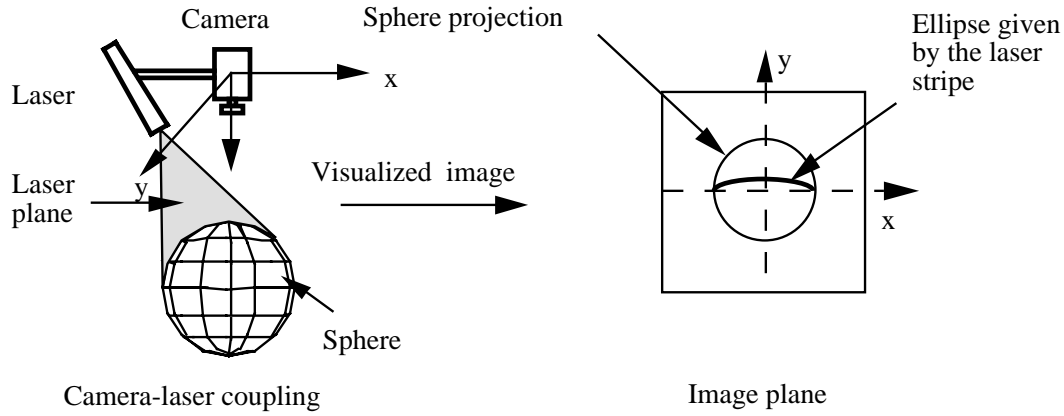


Figure 2: Camera-laser coupling with a sphere

Each laser stripe rigidly attached to the camera is characterized by a plane equation:

$$ax + by + cz + d = 0 \quad (20)$$

Considering a pinhole camera with unit focal length, a point  $\underline{x}(x, y, z)^T$  in 3D space projects into  $\underline{X} = (X \ Y \ 1)^T$  on the image frame with:

$$\underline{X} = \frac{1}{z} \underline{x} \quad (21)$$

By using (21) into (20) and (19), we can express the ellipse equation giving the projection in the image of the intersection between the sphere and the laser plane (see Figure 2). Obviously, the camera only detects the portion of the ellipse corresponding to the sphere top side. The equation of this ellipse is given by [4]:

$$E_1(\underline{X}, \underline{A}) = X^2 + A_1 Y^2 + 2A_2 XY + 2A_3 X + 2A_4 Y + A_5 = 0 \text{ with } \begin{cases} \underline{X} &= (X, Y) \\ \underline{A} &= (A_1, \dots, A_5) \end{cases} \quad (22)$$

where

$$\begin{cases} A_1 &= [b^2(z_0^2 + y_0^2 + x_0^2 - r^2) + 2bdy_0 + d^2]/A_0 \\ A_2 &= [ab(z_0^2 + y_0^2 + x_0^2 - r^2) + ady_0 + bdx_0]/A_0 \\ A_3 &= [ac(z_0^2 + y_0^2 + x_0^2 - r^2) + adz_0 + cdx_0]/A_0 \\ A_4 &= [bc(z_0^2 + y_0^2 + x_0^2 - r^2) + bdz_0 + cdy_0]/A_0 \\ A_5 &= [c^2(z_0^2 + y_0^2 + x_0^2 - r^2) + 2cdz_0 + d^2]/A_0 \end{cases} \quad (23)$$

$$\text{and } A_0 = a^2(z_0^2 + y_0^2 + x_0^2 - r^2) + 2adx_0 + d^2 \neq 0$$

### 3.1 $A_i$ Parameters

In this case, we have  $\underline{s} = (A_1, \dots, A_5)$  and  $\underline{p} = (a, b, c, d, x_0, y_0, z_0, r)$ . In order to establish, the interaction matrix related to these parameters, we have to compute the time variation of  $\underline{A}$  which can be expressed as follows:

$$\dot{\underline{A}} = \frac{\partial \underline{A}}{\partial \underline{p}} \frac{\partial \underline{p}}{\partial \underline{r}} \xi \quad (24)$$

In our conditions, we have  $\dot{a} = \dot{b} = \dot{c} = \dot{d} = \dot{r} = 0$ , since the laser plane is rigidly attached to the camera and the sphere keeps its shape.  $\underline{p}$  can thus be restricted to  $\underline{p} = \underline{m}_0 = (x_0, y_0, z_0)$ . We have  $\dot{\underline{m}}_0 = -\underline{T} - \underline{\Omega} \wedge \underline{m}_0$  which allows us to easily compute  $\frac{\partial \underline{p}}{\partial \underline{r}}$ . We obtain :

$$\frac{\partial \underline{p}}{\partial \underline{r}} = \begin{pmatrix} -1 & 0 & 0 & 0 & -z_0 & y_0 \\ 0 & -1 & 0 & z_0 & 0 & -x_0 \\ 0 & 0 & -1 & -y_0 & x_0 & 0 \end{pmatrix} \quad (25)$$

Moreover, we have from (23):

$$\frac{\partial \underline{s}}{\partial \underline{p}} = \frac{1}{A_0} \begin{pmatrix} 2b^2x_0 - \alpha A_1 & 2b^2y_0 + 2bd - 2a^2y_0A_1 & 2b^2z_0 - 2a^2z_0A_1 \\ 2abx_0 + bd - \alpha A_2 & 2aby_0 + ad - 2a^2y_0A_2 & 2abz_0 - 2a^2z_0A_2 \\ 2acx_0 + cd - \alpha A_3 & 2acy_0 - 2a^2y_0A_3 & 2acz_0 + ad - 2a^2z_0A_3 \\ 2bcx_0 - \alpha A_4 & 2bcy_0 + cd - 2a^2y_0A_4 & 2bcz_0 + bd - 2a^2z_0A_4 \\ 2c^2x_0 - \alpha A_5 & 2c^2y_0 - 2a^2y_0A_5 & 2c^2z_0 + 2cd - 2a^2z_0A_5 \end{pmatrix} \quad (26)$$

with  $\alpha = 2a^2x_0 + 2ad$ .

We finally obtain  $L_{\underline{A}}^T$  by the product of matrices  $\frac{\partial \underline{s}}{\partial \underline{p}}$  and  $\frac{\partial \underline{p}}{\partial \underline{r}}$ :

$$L_{\underline{A}}^T = -\frac{1}{A_0} \begin{pmatrix} 2b^2x_0 - 2A_1a^2x_0 - 2A_1ad & 2bd + 2b^2y_0 - 2A_1a^2y_0 & 2z_0(b^2 - A_1a^2) \\ -2A_2a^2x_0 - 2A_2ad + 2abx_0 + bd & a(d + 2by_0 - 2A_2ay_0) & 2az_0(b - A_2a) \\ -2A_3a^2x_0 - 2A_3ad + 2acx_0 + cd & 2ay_0(c - A_3a) & a(d + 2cz_0 - 2A_3az_0) \\ 2bcx_0 - 2A_4a^2x_0 - 2A_4ad & cd - 2A_4a^2y_0 + 2bcy_0 & bd + 2bcz_0 - 2A_4a^2z_0 \\ 2c^2x_0 - 2A_5a^2x_0 - 2A_5ad & 2y_0(c^2 - A_5a^2) & 2cd + 2c^2z_0 - 2A_5a^2z_0 \\ \\ -2bdz_0 & -2A_1adz_0 & 2d(A_1ay_0 + bx_0) \\ -adz_0 & dz_0(b - 2A_2a) & -d(by_0 - 2A_2ay_0 - ax_0) \\ ady_0 & -d(2A_3az_0 + ax_0 - cz_0) & -dy_0(c - 2A_3a) \\ d(by_0 - cz_0) & -d(2A_4az_0 + bx_0) & d(cx_0 + 2A_4ay_0) \\ 2cdy_0 & -2d(A_5az_0 + cx_0) & 2A_5ady_0 \end{pmatrix} \quad (27)$$

Knowing equation of the ellipse, we can compute other interaction matrices by using other representations in the image. One method uses the inertia moments of the ellipse, another one searches for points of discontinuity from the intersection of the ellipse given by the projection of the sphere in the image and the ellipse given by the intersection of the laser stripe and the sphere.

### 3.2 Inertia Moment

The parameters of the ellipse can be expressed from inertia moments  $m_{ij} = \sum_{X \in E} \sum_{Y \in E} X^i Y^j$  with  $i + j < 3$ . Inertia moments can be easily extracted from a digitized image.

From the equation (22), the natural representation of the ellipse fixing the centre coordinates  $(X_c, Y_c)$ , the lengths of axis  $(A, B)$  and the orientation  $(E)$  allows us to write:

$$\frac{(X - X_c + E(Y - Y_c))^2}{A^2(1 + E^2)} + \frac{(Y - Y_c - E(X - X_c))^2}{B^2(1 + E^2)} - 1 = 0 \quad (28)$$

with the following relations [4]:

$$\begin{aligned} X_c &= (A_1 A_3 - A_2 A_4) / (A_2^2 - A_1) \\ Y_c &= (A_4 - A_2 A_3) / (A_2^2 - A_1) \\ E &= (A_1 - 1 \pm \sqrt{(A_1 - 1)^2 + 4A_2^2}) / 2A_2 \\ A^2 &= 2(X_c^2 + 2A_2 X_c Y_c + A_1 Y_c^2 - A_5) / (1 + A_1 \pm \sqrt{(A_1 - 1)^2 + 4A_2^2}) \\ B^2 &= 2(X_c^2 + 2A_2 X_c Y_c + A_1 Y_c^2 - A_5) / (1 + A_1 \mp \sqrt{(A_1 - 1)^2 + 4A_2^2}) \end{aligned} \quad (29)$$

Knowing the parameters given by (29), the inertial moments are defined by:

$$\begin{cases} m_{00} = \pi AB ; m_{10} = \pi AB X_c ; m_{01} = \pi AB Y_c \\ m_{20} = \pi AB (A^2 + B^2 E^2) / 4(1 + E^2) + \pi AB X_c^2 \\ m_{11} = \pi AB E (A^2 - B^2) / 4(1 + E^2) + \pi AB X_c Y_c \\ m_{02} = \pi AB (A^2 E^2 + B^2) / 4(1 + E^2) + \pi AB Y_c^2 \end{cases} \quad (30)$$

Then, we choose as new representation of the ellipse, the parameters  $\underline{\mu} = (X_c, Y_c, \mu_{20}, \mu_{11}, \mu_{02})$  with [4]:

$$\begin{cases} X_c &= m_{10}/m_{00} \\ Y_c &= m_{01}/m_{00} \\ \mu_{20} &= 4(m_{20} - m_{00}X_c^2)/m_{00} = (A^2 + B^2E^2)/(1 + E^2) \\ \mu_{11} &= 4(m_{11} - m_{00}X_cY_c)/m_{00} = E(A^2 - B^2)/(1 + E^2) \\ \mu_{02} &= 4(m_{02} - m_{00}Y_c^2)/m_{00} = (A^2E^2 + B^2)/(1 + E^2) \end{cases} \quad (31)$$

From this set of parameters and the equation (29), we can find the relation between parameters  $\underline{\mu} = (X_c, Y_c, \mu_{20}, \mu_{11}, \mu_{02})$  and  $\underline{A} = (A_1 \dots A_5)$ , and we obtain:

$$\begin{cases} X_c &= (A_1A_3 - A_2A_4)/\Delta \\ Y_c &= (A_4 - A_2A_3)/\Delta \\ \mu_{20} &= -A_1K/\Delta \\ \mu_{11} &= -A_2K/\Delta \\ \mu_{02} &= -K/\Delta \end{cases} \quad (32)$$

$$\text{with } \Delta = A_2^2 - A_1 \text{ and } K = X_c^2 + 2A_2X_cY_c + A_1Y_c^2 - A_5$$

We can now construct the interaction matrix related to  $\underline{s} = \underline{\mu}$ . From (18), we have:

$$L_{\underline{\mu}}^T = \frac{\partial \underline{\mu}}{\partial \underline{A}} \frac{\partial \underline{A}}{\partial r} \quad (33)$$

where  $\frac{\partial \underline{A}}{\partial r}$  is nothing but the interaction matrix  $L_{\underline{A}}^T$  given in (27) and  $\frac{\partial \underline{\mu}}{\partial \underline{A}}$  can easily be determined from (31).

Using the following relation:

$$\begin{cases} A_1 &= \mu_{02}/\mu_{20} \\ A_2 &= \mu_{11}/\mu_{20} \\ A_3 &= -(X_c\mu_{20} + Y_c\mu_{11})/\mu_{20} \\ A_4 &= -(Y_c\mu_{02} + X_c\mu_{11})/\mu_{20} \\ A_5 &= [\mu_{11}^2 - \mu_{20}\mu_{02} + X_c^2\mu_{20} + 2\mu_{11}X_cY_c + \mu_{02}Y_c^2]/\mu_{20} \end{cases} \quad (34)$$

we finally obtain the interaction matrix  $L_{\underline{\mu}}^T = (L_{X_c}, L_{Y_c}, L_{\mu_{20}}, L_{\mu_{11}}, L_{\mu_{02}})^T$ , expressed with representation  $\underline{\mu}$  [11]:

$$\begin{aligned}
L_{X_c}^T &= -\frac{\mu_{20}}{K} [ \begin{array}{cc} (2\alpha x_0 \beta_1 + \mu_{02} d\alpha + dX_c \beta_1) & (2\alpha y_0 \beta_1 - \mu_{11} d\alpha + dY_c \beta_1) \\ (2\alpha z_0 \beta_1 + d\beta_1) & d(\mu_{11} \alpha z_0 + \beta_1 \lambda_1) \\ d(\mu_{02} \alpha z_0 - \beta_1 \lambda_2) & d(\alpha \gamma_1 + \beta_1 \lambda_3) \end{array} ] \\
L_{Y_c}^T &= -\frac{\mu_{20}}{K} [ \begin{array}{cc} (2\alpha x_0 \beta_2 - \mu_{11} d\alpha + dX_c \beta_2) & (2\alpha y_0 \beta_2 + \mu_{20} d\alpha + dY_c \beta_2) \\ (2\alpha z_0 \beta_2 + d\beta_2) & d(\beta_2 \lambda_1 - \mu_{20} \alpha z_0) \\ d(-\mu_{11} \alpha z_0 - \beta_2 \lambda_2) & d(\alpha \gamma_2 + \beta_2 \lambda_3) \end{array} ] \\
L_{\mu_{20}}^T &= -\frac{\mu_{20}}{K} [ \begin{array}{cc} 2(\mu_{20} \alpha (\alpha x_0 + X_c d) + \beta_2 (\beta_2 x_0 - \mu_{11} d)) & 2(\mu_{20} \alpha (\alpha y_0 + Y_c d) + \beta_2 (\beta_2 y_0 + \mu_{20} d)) \\ 2(\mu_{20} \alpha (\alpha z_0 + d) + \beta_2^2 z_0) & 2d(\mu_{20} \alpha \lambda_1 - \beta_2 \mu_{20} z_0) \\ -2d(\mu_{20} \alpha \lambda_2 + \beta_2 \mu_{11} z_0) & 2d(\mu_{20} \alpha \lambda_3 + \gamma_2 \beta_2) \end{array} ] \quad (35) \\
L_{\mu_{11}}^T &= -\frac{\mu_{20}}{K} [ \begin{array}{cc} [(2\mu_{11} \alpha (\alpha x_0 + X_c d) - 2\beta_1 \beta_2 x_0 + d(\mu_{11} \beta_1 - \mu_{02} \beta_2))] & [(2\mu_{11} \alpha (\alpha y_0 + Y_c d) - 2\beta_1 \beta_2 y_0 + d(\mu_{11} \beta_2 - \mu_{20} \beta_1))] \\ (2\mu_{11} \alpha (\alpha z_0 + d) - 2\beta_1 \beta_2 z_0) & d(2\mu_{11} \alpha \lambda_1 + z_0 (\mu_{20} \beta_1 - \mu_{11} \beta_2)) \\ -d(2\mu_{11} \alpha \lambda_2 - z_0 (\mu_{11} \beta_1 - \mu_{02} \beta_2)) & d(2\mu_{11} \alpha \lambda_3 - \beta_2 \gamma_1 - \beta_1 \gamma_2) \end{array} ] \\
L_{\mu_{02}}^T &= -\frac{\mu_{20}}{K} [ \begin{array}{cc} 2(\mu_{02} \alpha (\alpha x_0 + X_c d) + \beta_1 (\beta_1 x_0 + \mu_{02} d)) & 2(\mu_{02} \alpha (\alpha y_0 + Y_c d) + \beta_1 (\beta_1 y_0 - \mu_{11} d)) \\ 2(\mu_{02} \alpha (\alpha z_0 + d) + \beta_1^2 z_0) & 2d(\mu_{02} \alpha \lambda_1 + \beta_1 \mu_{11} z_0) \\ -2d(\mu_{02} \alpha \lambda_2 - \mu_{02} \beta_1 z_0) & 2d(\mu_{02} \alpha \lambda_3 + \gamma_1 \beta_1) \end{array} ] \\
\text{with } \left\{ \begin{array}{l} K = A_0(\mu_{11}^2 - \mu_{02} \mu_{20}) \\ \beta_1 = a\mu_{02} - \mu_{11}b \\ \beta_2 = b\mu_{20} - \mu_{11}a \\ \gamma_1 = -\mu_{11}x_0 - \mu_{02}y_0 \\ \gamma_2 = \mu_{20}x_0 + \mu_{11}y_0 \end{array} \right. & \text{and } \left\{ \begin{array}{l} \alpha = aX_c + bY_c + c \\ \lambda_1 = y_0 - Y_c z_0 \\ \lambda_2 = x_0 - X_c z_0 \\ \lambda_3 = Y_c x_0 - X_c y_0 \end{array} \right.
\end{aligned}$$

Let us remark that all these results were computed using MAPLE [5].

### 3.3 Point of Discontinuity Primitive

In order to elaborate the interaction matrix related to the point of discontinuity primitive, it is necessary to determine the expression of two ellipses:

- the ellipse given by the projection into the image of the intersection between the sphere and the laser plane (22);
- the ellipse given by the sphere projection onto the image.

This second ellipse (a circle if the sphere is centered in the image) can be represented in a similar way to the first one:

$$E_2(\underline{X}, \underline{B}) = X^2 + B_1 Y^2 + 2B_2 XY + 2B_3 X + 2B_4 Y + B_5 = 0 \text{ with } \begin{cases} \underline{X} = (X, Y) \\ \underline{B} = (B_1, \dots, B_5) \end{cases} \quad (36)$$

where [11]:

$$\begin{cases} B_1 &= [r^2 - x_0^2 - z_0^2]/B_0 \\ B_2 &= [x_0 y_0]/B_0 \\ B_3 &= [x_0 z_0]/B_0 \\ B_4 &= [y_0 z_0]/B_0 \\ B_5 &= [r^2 - x_0^2 - y_0^2]/B_0 \end{cases} \quad (37)$$

and  $B_0 = r^2 - y_0^2 - z_0^2$

Theoretically, and knowing these two expressions of the ellipse, we have two points of discontinuity  $\underline{X}_e = (X_e, Y_e)$  which correspond to their intersections. Then we can determine the interaction matrix related to each point of discontinuity. We compute the time variation of the expression  $E_1(\underline{X}, \underline{A})$  and  $E_2(\underline{X}, \underline{B})$ . We have:

$$\dot{E}_1 = \frac{\partial E_1}{\partial \underline{X}_e} \dot{\underline{X}}_e + \frac{\partial E_1}{\partial \underline{A}} \dot{\underline{A}} = 0 \quad (38)$$

$$\dot{E}_2 = \frac{\partial E_2}{\partial \underline{X}_e} \dot{\underline{X}}_e + \frac{\partial E_2}{\partial \underline{B}} \dot{\underline{B}} = 0 \quad (39)$$

We thus obtain a linear system with  $\dot{X}_e, \dot{Y}_e$  as unknowns. The resolution of this system gives the interaction matrix  $L_{\underline{X}_e}^T$  of the point of discontinuity, knowing:

- the expressions  $A_i$  given by (23),  $\dot{A}_i$  given by (27),  $B_i$  given by (37) and  $\dot{B}_i$  obtained in a similar manner to the interaction matrix of the parameters  $A_i$ . These expressions depend on 3D scene parameters and the sphere velocity  $(\underline{T}, \underline{\Omega})$  with respect to the sensor.
- the coordinates  $X_e, Y_e$  of the point of discontinuity extracted after each image acquisition.

After some simplifications, the interaction matrix can be expressed by:

$$\dot{\underline{X}}_e = \begin{pmatrix} \dot{X}_e \\ \dot{Y}_e \end{pmatrix} = L_{\underline{X}_e}^T \cdot \xi \quad (40)$$

where

$$\begin{aligned} L_{X_e}^T &= -\frac{1}{K} \begin{bmatrix} \lambda_2 \beta (x_0 \beta + d X_e) + \lambda_4 (Y_e \alpha_3 - \alpha_1) & \lambda_2 \beta (y_0 \beta + d Y_e) + \lambda_4 (-X_e \alpha_3 - \alpha_2) \\ \lambda_2 \beta (z_0 \beta + d) + \lambda_4 (X_e \alpha_1 + Y_e \alpha_2) & \alpha_2 (\lambda_4 \gamma - d \beta \lambda_2) \\ -\alpha_1 (\lambda_4 \gamma - d \beta \lambda_2) & -\alpha_3 (\lambda_4 \gamma - d \beta \lambda_2) \end{bmatrix} \\ L_{Y_e}^T &= -\frac{1}{K} \begin{bmatrix} -\lambda_1 \beta (x_0 \beta + d X_e) - \lambda_3 (Y_e \alpha_3 - \alpha_1) & -\lambda_1 \beta (y_0 \beta + d Y_e) - \lambda_3 (-X_e \alpha_3 - \alpha_2) \\ -\lambda_1 \beta (z_0 \beta + d) - \lambda_3 (X_e \alpha_1 + Y_e \alpha_2) & \alpha_2 (\lambda_1 \beta d - \gamma \lambda_3) \\ -\alpha_1 (\lambda_1 \beta d - \gamma \lambda_3) & -\alpha_3 (\lambda_1 \beta d - \gamma \lambda_3) \end{bmatrix} \end{aligned} \quad (41)$$

$$\text{with } \begin{cases} \alpha_1 &= X_e z_0 - x_0 \\ \alpha_2 &= Y_e z_0 - y_0 \\ \alpha_3 &= Y_e x_0 - X_e y_0 \\ \beta &= X_e a + Y_e b + c \\ \gamma &= Y_e y_0 + X_e x_0 + z_0 \end{cases} \quad \text{and} \quad \begin{cases} \lambda_1 &= B_0 (X_e + B_2 Y_e + B_3) \\ \lambda_2 &= B_0 (B_1 Y_e + B_2 X_e + B_4) \\ \lambda_3 &= A_0 (X_e + A_2 Y_e + A_3) \\ \lambda_4 &= A_0 (A_1 Y_e + A_2 X_e + A_4) \\ K &= 2(\lambda_1 \lambda_4 - \lambda_3 \lambda_2) \end{cases}$$

## 4 Results

The chosen task consists of positioning the camera with respect to a sphere in such way that the projection in the image gives a centered circle ( $x_0 = y_0 = 0, z_0 = z^*$ ). The experimentations have been performed with a sphere of radius  $3cm$ , the desired distance  $z^*$ , between camera and object, being fixed to  $30cm$ . We use the different parametrizations presented in the previous section i.e., the  $A_i$  parameters, the inertia moments parameters and the points of discontinuity parameters obtained by the intersection of the sphere ellipsis and the laser plane ellipsis [11]. First of all, we briefly present the experimental cell built in our laboratory.

### 4.1 Hardware System Organization

For visual servoing applications, both the image analysis and the robot control must be performed continuously, in real-time, and with only modest computing hardware. To meet the practical constraints of cost and real-time implementation, we have built a laboratory experimental cell based on the following considerations.

- The flow of data through the system has to be organized to minimize *delays* and *storage*. Therefore, we opted for an integrated robot-vision control architecture.
- Sensor-motor cooperation : In order to ensure the stability of the control, the system has to be sampled at an adequate rate. The results computed by the image processing algorithm have to be furnished at a fixed rate to the robot controller (the results must be consistent with the given input image). Moreover, the processing time must be consistent with the sampling period of the robot controller. For visual servoing tasks, based on practical considerations (acquisition time of an image, processing power of current available computers and hardware versus amount of processing real-time experiments), we suggest a minimal rate of 10 images per second. This means that visual analysis has to be flow-through, avoiding iterative loops, with storage of parameters of visual informations.

The functional elements of our system consists of a robot control and vision processing modules (see Figure 3). They take place concurrently in an integrated robot and vision controller organized around the VME bus using a multi-processor architecture. In order to master the timing of the processing, we use the VERSADOS operating system, a multi-tasking and multi-process real-time operating system which controls the timing and sharing of the various vision processing and control tasks. The different tasks communicate with each other by interruptions under the supervision of a master controller. For visual servoing purposes, only a limited number of image features have to be detected and extracted at video rate. As we know the information to extract, the pre-processing stage (processing on the data flow) seems to be the best level for hardware implementation. Finally, we note that the sensor is positioned on the last body of a robot with five degrees of freedom.



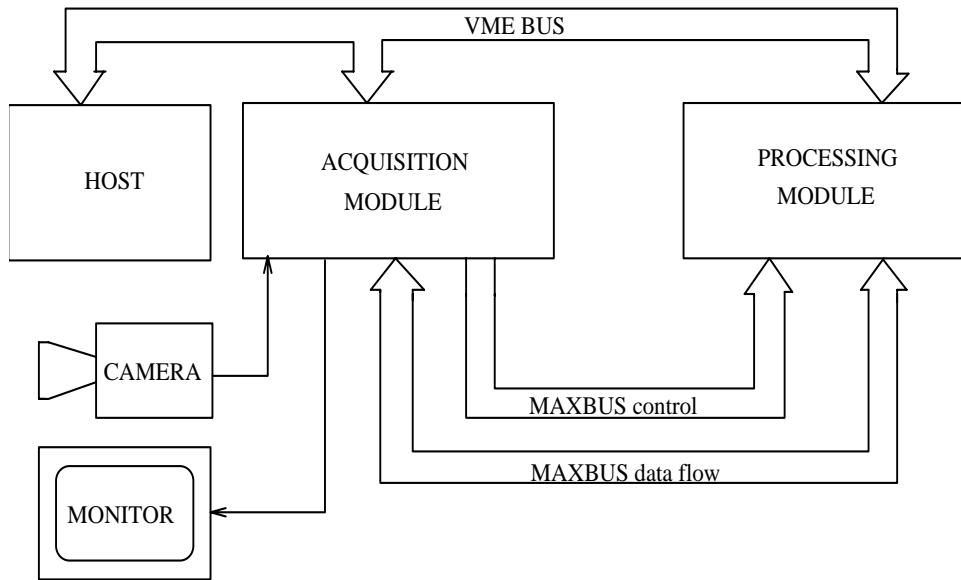


Figure 3: Image processing module

## 4.2 Results Using the $A_i$ Parameters

We first realize the positioning task using the  $A_i$  parameters. Both laser planes have been calibrated in order to produce two orthogonal ellipse centered in the image at the desired position. These particular configurations simplify the relations defined in Section 3. We first have:

$$\text{Laser 1: } \begin{cases} a_{l1} = 0 \\ b_{l1} \neq 0 \\ c_{l1} = 1 \\ d_{l1} = -(z^{2*} - r^{2*})/z^* \end{cases} \quad \text{Laser 2: } \begin{cases} a_{l2} \neq 0 \\ b_{l2} = 0 \\ c_{l2} = 1 \\ d_{l2} = -(z^{2*} - r^{2*})/z^* \end{cases} \quad (42)$$

With two laser stripes, we can choose  $\underline{s} = (A_{11} \ A_{21} \ A_{31} \ A_{41} \ A_{51} \ A_{12} \ A_{22} \ A_{32} \ A_{42} \ A_{52})$  where  $A_{ij}$  is the parameter  $A_i$  corresponding to the ellipse  $j$ . At a desired position, we can compute  $\underline{s}^*$  using (23):

$$\underline{s}^* = (A_{11}^* \ A_{21}^* \ A_{31}^* \ A_{41}^* \ A_{51}^* \ A_{12}^* \ A_{22}^* \ A_{32}^* \ A_{42}^* \ A_{52}^*)$$

with:

$$\begin{cases} A_{11}^* = [b_{l1}^2(z^* - r^2) + d_{l1}^2]/A_{01}^* & A_{12}^* = d_{l2}^2/A_{02}^* \\ A_{21}^* = 0 & A_{22}^* = 0 \\ A_{31}^* = 0 & A_{32}^* = [a_{l2}c_{l2}(z^* - r^2) + a_{l2}d_{l2}z^*]/A_{02}^* \\ A_{41}^* = [b_{l1}c_{l1}(z^* - r^2) + b_{l1}d_{l1}z^*]/A_{01}^* & A_{42}^* = 0 \\ A_{51}^* = [c_{l1}^2(z^* - r^2) + 2c_{l1}d_{l1}z^* + d_{l1}^2]/A_{01}^* & A_{52}^* = [c_{l2}^2(z^* - r^2) + 2c_{l2}d_{l2}z^* + d_{l2}^2]/A_{02}^* \\ A_{01}^* = d_{l1}^2 & A_{02}^* = a_{l2}^2(z^* - r^2) + d_{l2}^2 \end{cases}$$

The interaction matrix related to  $\underline{s} = \underline{s}^*$  is obtained from (27):

$$L_{|\underline{s}=\underline{s}^*}^T = - \begin{pmatrix} 0 & \frac{2b_1 d_1}{A_{01}^*} & \frac{2b_1^2 z^*}{A_{01}^*} & \frac{-2b_1 d_1 z^*}{A_{01}^*} & 0 & 0 \\ \frac{b_1 d_1}{A_{01}^*} & 0 & 0 & 0 & \frac{b_1 d_1 z^*}{A_{01}^*} & 0 \\ \frac{c_1 d_1}{A_{01}^*} & 0 & 0 & 0 & \frac{c_1 d_1 z^*}{A_{01}^*} & 0 \\ 0 & \frac{c_1 d_1}{A_{01}^*} & \frac{b_1 (2c_1 z^* + d_1)}{A_{01}^*} & \frac{-c_1 d_1 z^*}{A_{01}^*} & 0 & 0 \\ 0 & 0 & \frac{2c_1 (c_1 z^* + d_1)}{A_{01}^*} & 0 & 0 & 0 \\ \frac{-2A_{12}^* a_2 d_2}{A_{02}^*} & 0 & \frac{-2A_{12}^* a_2^2 z^*}{A_{02}^*} & 0 & \frac{-2A_{12}^* a_2 d_2 z^*}{A_{02}^*} & 0 \\ 0 & \frac{a_2 d_2}{A_{02}^*} & 0 & \frac{-a_2 d_2 z^*}{A_{02}^*} & 0 & 0 \\ \frac{-d_2 (2A_{32}^* a_2 - c_2)}{A_{02}^*} & 0 & \frac{-a_2 (-2c_2 z^* - d_2 + 2A_{32}^* a_2 z^*)}{A_{02}^*} & 0 & \frac{-d_2 z^* (2A_{32}^* a_2 - c_2)}{A_{02}^*} & 0 \\ 0 & \frac{c_2 d_2}{A_{02}^*} & 0 & \frac{-c_2 d_2 z^*}{A_{02}^*} & 0 & 0 \\ \frac{-2A_{52}^* a_2 d_2}{A_{02}^*} & 0 & \frac{2c_2^2 z^* + 2c_2 d_2 - 2A_{52}^* a_2^2 z^*}{A_{02}^*} & 0 & \frac{-2A_{52}^* a_2 d_2 z^*}{A_{02}^*} & 0 \end{pmatrix} \quad (43)$$

In this case, the rank of  $L_{|\underline{s}=\underline{s}^*}^T$  is 3. It means that 3 camera degrees of freedom, correctly chosen using the form of the interaction matrix, are sufficient to achieve this task. Indeed, the kernel of this matrix is given by :

$$Ker(L_{|\underline{s}=\underline{s}^*}^T) = \begin{pmatrix} -z^* & 0 & 0 \\ 0 & z^* & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (44)$$

To perform this task, we have used the camera translational velocities  $T_x, T_y$  and  $T_z$ . In that case, the corresponding interaction matrix is composed of the first three columns of the general form given in (43) and the matrix  $W$  can be chosen as  $\mathbf{I}_3$  (proving the condition (7)). So, the global task function (5) is given by :

$$\underline{e} = W^+ W L_{|\underline{s}=\underline{s}^*}^{T+} (\underline{s}(r) - \underline{s}^*) = L_{|\underline{s}=\underline{s}^*}^{T+} (\underline{s}(r) - \underline{s}^*) \quad (45)$$

Let us note that, because of the particular configuration of  $L_{|\underline{s}=\underline{s}^*}^T$ ,  $\Omega_y$  could be used instead of  $V_x$  and  $\Omega_x$  instead of  $V_y$ .

We present simulation results on the Figure 4, with noises corresponding to errors in the geometric model of the robot and to the ellipsis parameter extraction error owing to the image processing (2% white noise on the image coordinates and on the camera location).

The different windows show (see Figure 4):

- (a): the target image (configuration of the ellipse in the target image);
- (b): the initial image observed by the camera before the visual servoing;
- (c): the behaviour of each component of the control vector during the visual servoing (translational velocity  $\underline{T}_x, \underline{T}_y, \underline{T}_z$ );

- (d): the evolution of the error :  $||s - s^*||$ ;
- (e): the behaviour of each point of discontinuity during the task.

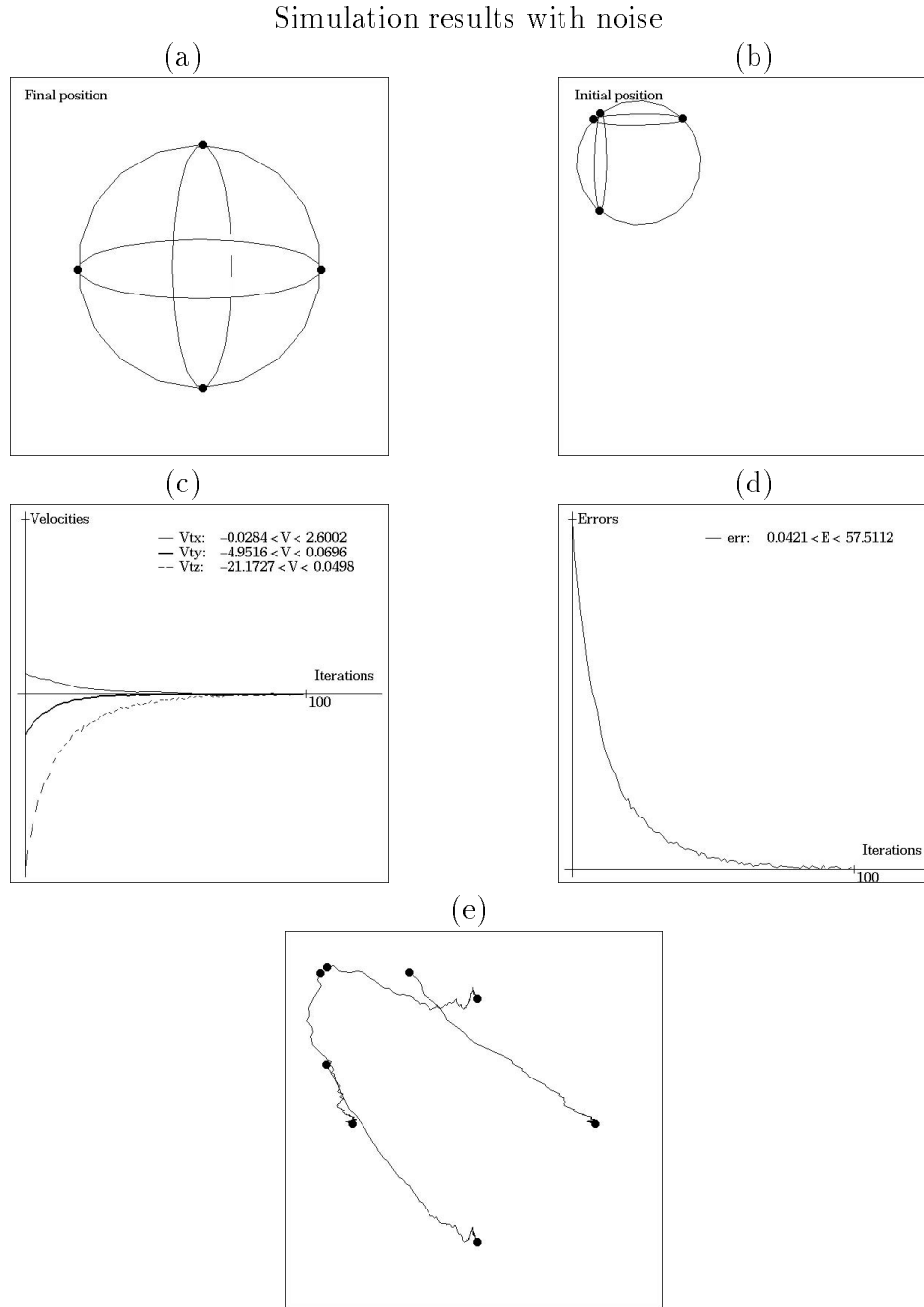


Figure 4: Use of the  $A_i$  parameters in the control law

We can observe in Figure 4 that the convergence to the desired image target is correctly performed. The stability and the robustness of the control scheme have been proved in several

simulation conditions. Note that the noise introduced on the measurements and on the robot locations brings little perturbation to the system.

### 4.3 Results Using the Inertia Moments

We can realize the same positioning task from parameters given by the inertia moments. We can choose  $\underline{s} = (X_{c_1}, Y_{c_1}, \mu_{20_1}, \mu_{11_1}, \mu_{02_1}, X_{c_2}, Y_{c_2}, \mu_{20_2}, \mu_{11_2}, \mu_{02_2})$ , where index  $i$  corresponds to the ellipse given by the laser plane  $i$ .

At the equilibrium position, and using (31), we can compute  $\underline{s}^* = (X_{c_1}^*, Y_{c_1}^*, \mu_{20_1}^*, \mu_{11_1}^*, \mu_{02_1}^*, X_{c_2}^*, Y_{c_2}^*, \mu_{20_2}^*, \mu_{11_2}^*, \mu_{02_2}^*)$  with:

$$\begin{cases} X_{c_1}^* = 0 \\ Y_{c_1}^* = 0 \\ \mu_{20_1}^* = r^{2*}/(b_{l1}^2 z^{2*} + z^{2*} - r^{2*}) \\ \mu_{11_1}^* = 0 \\ \mu_{02_1}^* = r^{2*}/(z^{2*} - r^{2*}) \end{cases} \quad \text{and} \quad \begin{cases} X_{c_2}^* = 0 \\ Y_{c_2}^* = 0 \\ \mu_{20_2}^* = r^{2*}/(z^{2*} - r^{2*}) \\ \mu_{11_2}^* = 0 \\ \mu_{02_2}^* = r^{2*}/(a_{l2}^2 z^{2*} + z^{2*} - r^{2*}) \end{cases} \quad (46)$$

By using (35), the expression of the interaction matrix related to  $\underline{s} = \underline{s}^*$  can be easily obtained:

$$L_{|\underline{s}=\underline{s}^*}^T = - \begin{pmatrix} \frac{1}{\alpha} & 0 & 0 & 0 & \frac{z^*}{\alpha} & 0 \\ 0 & \frac{\mu_{20_1}^*}{\alpha \mu_{02_1}^*} & -\beta \frac{(2z^* - \alpha)}{\alpha^2 \mu_{02_1}^*} & -\frac{\mu_{20_1}^* z^*}{\alpha \mu_{02_1}^*} & 0 & 0 \\ 0 & 2 \frac{\mu_{20_1}^* \beta}{\alpha \mu_{02_1}^*} & -2 \frac{[\mu_{20_1}^* (z^* - \alpha) + \beta^2 z^*]}{\alpha^2 \mu_{02_1}^*} & -2 \frac{\mu_{20_1}^* \beta z^*}{\alpha \mu_{02_1}^*} & 0 & 0 \\ -\frac{\beta}{\alpha} & 0 & 0 & 0 & -\frac{\beta z^*}{\alpha} & 0 \\ 0 & 0 & -2 \frac{(z^* - \alpha)}{\alpha^2} & 0 & 0 & 0 \\ \frac{\alpha}{A_0} & 0 & -\gamma \frac{(2z^* - \alpha)}{A_0 \mu_{02_2}^*} & 0 & \frac{\alpha z^*}{A_0} & 0 \\ 0 & \frac{\mu_{20_2}^* \alpha}{A_0 \mu_{02_2}^*} & 0 & -\frac{\mu_{20_2}^* \alpha z^*}{A_0 \mu_{02_2}^*} & 0 & 0 \\ 0 & 0 & -2 \frac{\mu_{20_2}^* (z^* - \alpha)}{A_0 \mu_{02_2}^*} & 0 & 0 & 0 \\ 0 & -\frac{\alpha \mu_{20_2}^* \gamma}{A_0 \mu_{02_2}^*} & 0 & \frac{\alpha \mu_{20_2}^* \gamma z^*}{A_0 \mu_{02_2}^*} & 0 & 0 \\ 2 \frac{\alpha \gamma}{A_0} & 0 & -2 \frac{[\mu_{02_2}^* (z^* - \alpha) + \gamma^2 z^*]}{A_0 \mu_{02_2}^*} & 0 & 2 \frac{\alpha \gamma z^*}{A_0} & 0 \end{pmatrix} \quad (47)$$

$$\text{with} \begin{cases} \alpha = (z^{2*} - r^2)/z^* \\ \beta = a_{l1} \mu_{20_1}^* \\ \gamma = b_{l2} \mu_{20_2}^* \\ A_{02} = a_{l2}^2 (z^{2*} - r^2) + \alpha^2 \end{cases}$$

This matrix has the same kernel, given by (44), than using the  $A_i$  parameters. In this case, we have used camera velocity vectors  $T_z, \Omega_x$  and  $\Omega_y$  to accomplish the task. The expression of the corresponding task function and control law can be obtained in a similar manner as stated previously.

Figure 5 presents simulation results in the presence of noise.

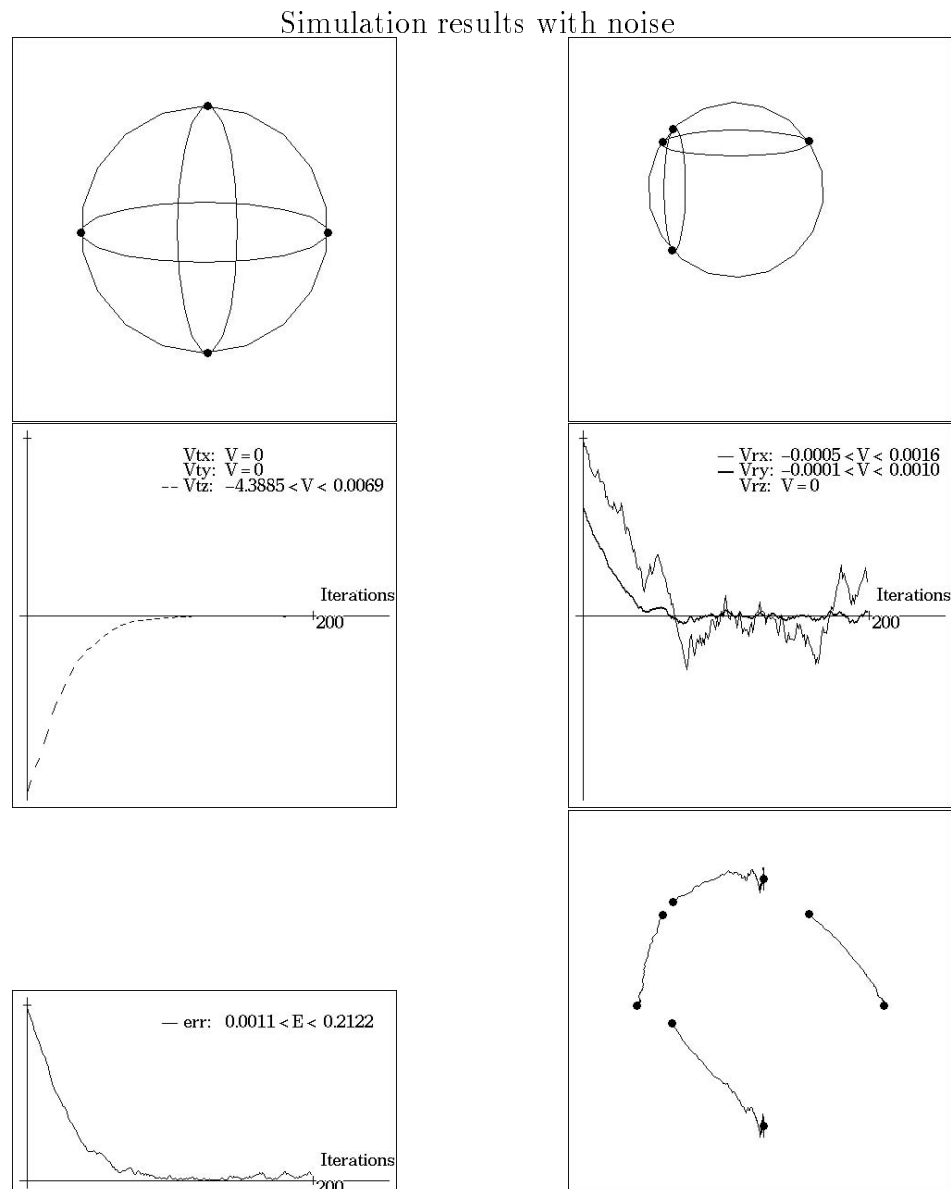


Figure 5: Use of the inertia moments in the control law

#### 4.4 Results Using the Points of Discontinuity

The same positioning task has been realized in our experimental cell by using points of discontinuity of the ellipse. Moreover, we use a five degrees of freedom manipulator which does not provide the rotation  $\Omega_x$ .

Using two laser stripes to perform this task, we have  $\underline{s} = (X_1, Y_1, \dots, X_4, Y_4)$ , where the first two points belong to the laser plane 1 ( $a_{l1} = 0$ ) and the last two points belong to the laser plane 2 ( $b_{l2} = 0$ ). At the equilibrium position, the sphere is still centered in the image, thus  $\underline{s}^* = (X^*, 0, -X^*, 0, 0, Y^*, 0, -Y^*)$  with  $X^* = Y^* = r/\sqrt{z^{*2} - r^2}$ .

The rank of the interaction matrix, related to  $\underline{s} = \underline{s}^*$ , is again 3. The matrix is obtained from (41):

$$L_{\underline{s}=\underline{s}^*}^T = -\frac{1}{K} \begin{pmatrix} z^* & 0 & -X^*z^* & 0 & z^{*2} & 0 \\ -\frac{X^*K}{2b_{l1}} & 0 & \frac{z^{*2}+r^2}{2b_{l1}} & 0 & -\frac{X^*Kz^*}{2b_{l1}} & 0 \\ z^* & 0 & X^*z^* & 0 & z^{*2} & 0 \\ \frac{X^*K}{2b_{l1}} & 0 & \frac{z^{*2}+r^2}{2b_{l1}} & 0 & \frac{X^*Kz^*}{2b_{l1}} & 0 \\ 0 & -\frac{Y^*K}{2a_{l2}} & \frac{z^{*2}+r^2}{2a_{l2}} & \frac{Y^*Kz^*}{2a_{l2}} & 0 & 0 \\ 0 & z^* & -Y^*z^* & -z^{*2} & 0 & 0 \\ 0 & \frac{Y^*K}{2a_{l2}} & \frac{z^{*2}+r^2}{2a_{l2}} & -\frac{Y^*Kz^*}{2a_{l2}} & 0 & 0 \\ 0 & z^* & Y^*z^* & -z^{*2} & 0 & 0 \end{pmatrix} \quad (48)$$

$$\text{with } K = z^{*2} - r^2$$

In this application, the processing applied to the image consists of a simple direct thresholding which allows to keep only the information corresponding to the over light area due to the laser stripe projection onto the scene. Then it is easy to extract the points of discontinuity of the ellipse.

Results obtained in our experimental cell using the 3 translational degrees of freedom are shown in Figure 6. They show the stability and the exponential convergence of the control law. The convergency to the desired configuration of the target in the image is performed. These experimental results and those obtained in simulation, with other parameters, possess approximately the same behaviour.

Finally, we present experimental results of the same positioning task with a secondary task which consists in a rotation  $\Omega_y$  independent of the visual servoing. In order to achieve this secondary task, it is necessary to choose the matrix  $W$  according to the condition  $Ker(W) = Ker(L_{\underline{s}=\underline{s}^*}^T)$ . Intuitively, it seems difficult to define this matrix  $W$  from the expression of the interaction matrix. That is why we express this matrix from different rows of  $L_{\underline{s}=\underline{s}^*}^T$ . For instance,  $W$ , which has to be a  $3 \times 6$  matrix of full rank 3, may be chosen as:

$$W = \frac{1}{K} \begin{pmatrix} z^* & 0 & -X^*z^* & 0 & z^{*2} & 0 \\ -\frac{X^*K}{2b_{l1}} & 0 & \frac{z^{*2}+r^2}{2b_{l1}} & 0 & -\frac{X^*Kz^*}{2b_{l1}} & 0 \\ 0 & -\frac{Y^*K}{2a_{l2}} & \frac{z^{*2}+r^2}{2a_{l2}} & \frac{Y^*Kz^*}{2a_{l2}} & 0 & 0 \end{pmatrix} \quad (49)$$

Irisa

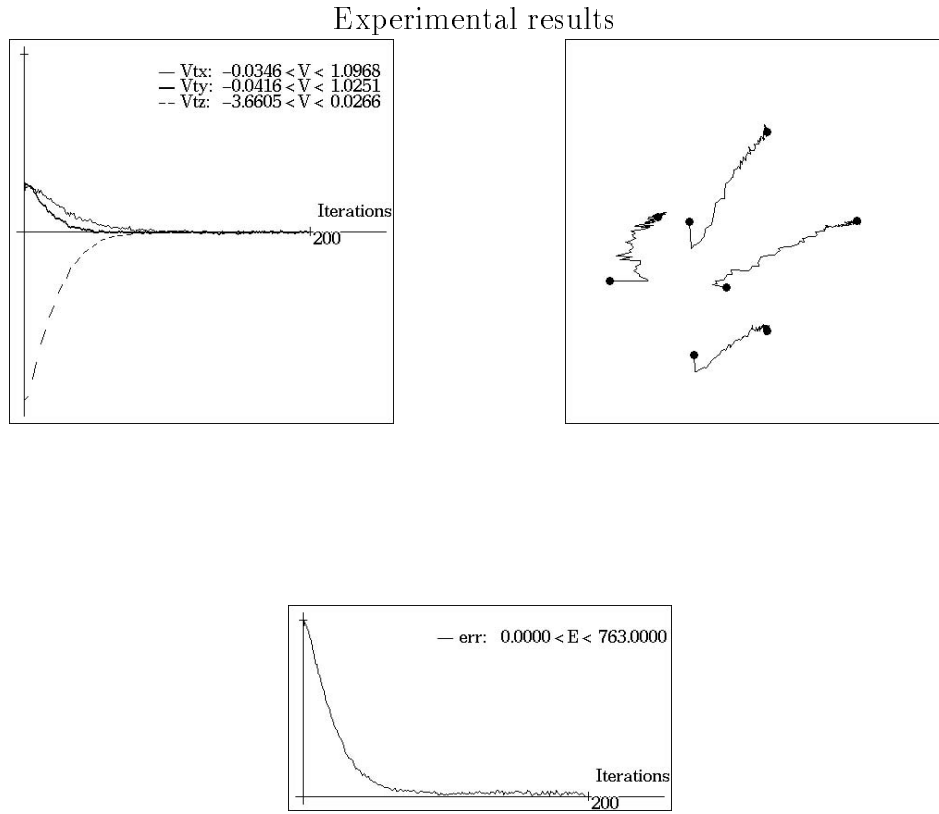


Figure 6: Use of points of discontinuity

Then, the global task function  $\underline{e}$  can be expressed as:

$$\underline{e} = W^+ C(\underline{s} - \underline{s}^*) + (\mathbf{I}_6 - W^+ W) g_s^T \quad (50)$$

where  $W^+$  is the pseudo inverse of  $W$ ,  $C = W L_{|\underline{s}=\underline{s}^*}^{T+}$  and

$$(\mathbf{I}_6 - W^+ W) = \frac{1}{(z^{2*} + 1)} \begin{pmatrix} z^{2*} & 0 & 0 & 0 & -z^* & 0 \\ 0 & z^{2*} & 0 & z^* & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & z^* & 0 & 1 & 0 & 0 \\ -z^* & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (51)$$

The secondary task, which corresponds to the rotation of the camera around  $y$  axis at a constant velocity  $\omega$ , comes down to minimize the following secondary cost  $h_s$ :

$$h_s = \frac{1}{2} \beta (\theta_y(t) - \theta_y(0) - \omega t)^2 \quad (52)$$

where  $\theta_y(0)$  is the initial orientation  $y$  of the camera,  $\theta_y(t)$  is the current value and  $\beta$  is a positive scalar weight.

We can compute:

$$\underline{g}_s^T = \frac{\partial h_s}{\partial \underline{r}} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \beta(\theta_y(t) - \theta_y(0) - \omega t) \\ 0 \end{pmatrix}, \quad \frac{\partial \underline{g}_s^T}{\partial t} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\beta\omega \\ 0 \end{pmatrix} \quad (53)$$

Therefore, we can determine the control law applied to the camera:

$$\xi_c = -\lambda W^+ C(\underline{s} - \underline{s}^*) - \frac{\lambda\beta}{(z^{2*} + 1)} \begin{pmatrix} -z^*(\theta_y(t) - \theta_y(0) - \omega t) \\ 0 \\ 0 \\ 0 \\ (\theta_y(t) - \theta_y(0) - \omega t) \\ 0 \end{pmatrix} + \frac{\beta}{(z^{2*} + 1)} \begin{pmatrix} -z^*\omega \\ 0 \\ 0 \\ 0 \\ \omega \\ 0 \end{pmatrix} \quad (54)$$

Thus, we choose  $\beta = (z^{2*} + 1)$  in order that the rotational velocity around  $y$  axis has the desired value  $\omega$  when  $\underline{e} = 0$ . We can note that the secondary task involves the desired rotational velocity  $\omega$  around  $y$  axis but also implies a translational velocity along  $x$  axis. This translational motion compensates the rotational motion in order that the camera turns around the sphere without any variation in the image. Experimental results of this task are presented in Figure 7. The amplitude  $\omega$  of the rotational motion has been fixed to 1 degree/s and  $\lambda = 0.1$ . The algorithm is decomposed as follows:

- positioning until the desired situation for 100 iterations,
- rotational motion  $\Omega_y$  for 100 iterations,
- opposite rotational motion  $\Omega_y$  for 200 iterations, e.t.c.

We can see from Figure 7 that the camera moves in order that the desired visual data is reached in the image after the first 100 iterations. Additionally, the visual data remains unchanged during the secondary task accomplishment (the error remains around zero).

We can also remark that whatever the used parametrization type, simulation and experiment be, the results have approximately the same behaviour.

## 5 Conclusion

In some applications, the passive vision may turn out to be restrictive due to the difficulty of extracting “useful” information of the image. That is why a particular sensor, constituted by the coupling of a camera and laser stripe, was chosen. Indeed, with such a sensor, the image processing is significantly reduced, and moreover the primitives in the image are relatively straightforward.



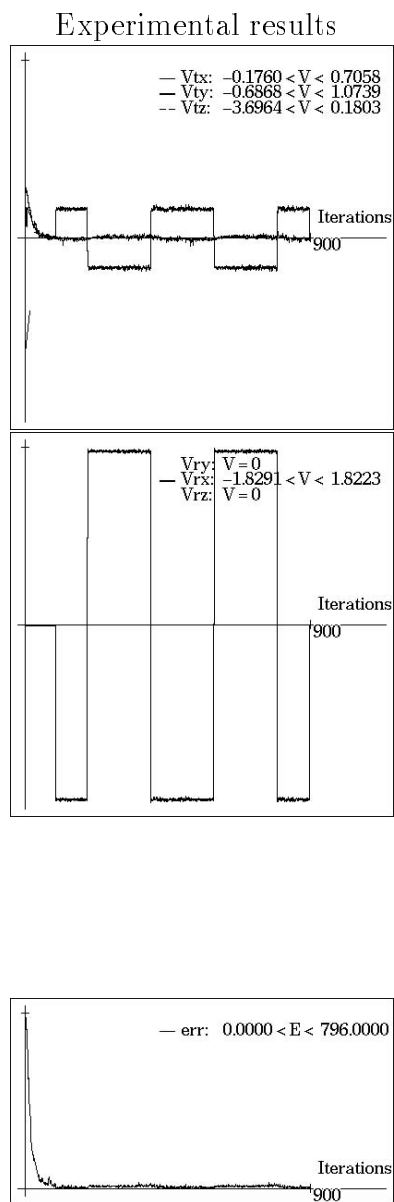


Figure 7: Use of points of discontinuity (with a secondary task)

In this paper, we have presented a general method for modelling visual features using this useful sensor, and applied this method to a spherical scene case. This modelling establishes a relation between the variation of visual information and the various motions of the sensor. Then, we integrated these works under the *task function approach* which allows the achievement of positioning robotics tasks with good robustness and stability conditions. Finally, in simulation and in our experimental cell, results were presented for the positioning task with respect to a sphere.

We have demonstrated the various advantages of a camera-laser coupling. Nevertheless, this sensor has some constraints. This coupling involves some restrictions in the laser stripe projection onto the scene. It is necessary to choose the most favourable situation of the laser stripe in order to achieve a robotics task in conditions of best stability. Moreover, a calibration step is essential in order to compute each parameter of a laser plane. This calibration step is necessary in order to compute the desired position to be reached in the image (this could also be done by a learning approach) but, as shown in [8], calibration parameters are not sensitive for the stability, robustness and convergence of the visual servoing.

In these approaches, the sampling rate of the robot closed loop control is directly given by the image processing time. Hence, for this reason, an original vision system (WINDIS: WINdow DIStributor) based on a modular and parallel architecture concept is developed [14]. This permits us to design control applications based on vision with real-time capabilities.

## References

- [1] J.E. Agapakis : *Approaches for Recognition and Interpretation of Workpiece Surface Features Using Structured Lighting*, Int Journal of Robotics Research, Vol.9, n.5, pp. 3-16, October 1990.
- [2] G.J. Agin : *Real-time control of a robot with a mobile camera*, 9th Int. Symp. on Industrial Robots, Washington D.C., pp. 233-246, March 1979.
- [3] J.S. Albus, A.J. Barbera, M.L. Fitzgerald : *Hierarchical Control for Sensory Interactive Robots*, Proc. of 11<sup>th</sup> Int. Symp. on Industrial Robotics, pp. 497-505, Tokyo, Japan, October 1981.
- [4] F. Chaumette : *La relation vision-commande : théorie et application à des tâches robotiques*, Phd Thesis IRISA/INRIA, Rennes, France, July 1990.
- [5] B.W. Char, K.O. Geddes, G.H. Bonnet, M.B. Monagan, S.M. Watt : *MAPLE : Reference Manual*, 5th Edition, March 1988.
- [6] B. Espiau, C. Samson : *Sensory based control. Robustness issues and modelling technics. Application to proximity sensing*, NATO Workshop on Kinematic and dynamic Issues in Sensor Based Control, Pisa, October 1987.
- [7] B. Espiau, F. Chaumette, P. Rives : *A new Approach to Visual Servoing in Robotics*, in IEEE Trans. on Robotics and Automation, vol. 8, n. 3, pp. 313-326, June 1992.

- [8] B. Espiau : *Effect of Camera Calibration Errors on Visual Servoing in Robotics*, Proc. of the 3rd Int. Symposium on Experimental Robotics, Kyoto, Japan, October 1993.
- [9] J.T. Feddema, C.S.G. Lee, O.R. Mitchell : *Automatic Selection of Image Features for Visual Servoing of a Robot Manipulator*, Proc. of the IEEE Int. Conf. on Robotics and Automation, pp 832-837, Scottsdale, Arizona, USA, May 1989.
- [10] J.T. Feddema, O.R. Mitchell : *Vision-Guided Servoing with Feature-Based Trajectory Generation*, IEEE Trans. on Robotics and Automation, Vol. 5, n. 5, pp. 691-700, October 1989.
- [11] G. Motyl : *Couplage d'une caméra et d'un faisceau laser en commande référencée vision*, PhD Thesis, Clermont-Ferrand, France, September 1992.
- [12] G. Motyl, F. Chaumette, J. Gallice : *A Camera and Laser Stripe in Sensor Based Control*, Second International Symposium on Measurement and Control in Robotics, AIST Tsukuba Research Center, Japan, pp. 685-692, November 1992.
- [13] G. Motyl, P. Martinet, J. Gallice : *Visual Servoing with Respect to a Target Sphere Using a Camera/Laser-Stripe Sensor*, in proceedings'93 International Conference on Advanced Robotics, Tokyo, Japan, pp. 591-596, November 1993.
- [14] P. Rives, J.J. Borrelly, J. Gallice, P. Martinet : *A Versatile Parallel Architecture for Vision Servoing Applications*, Workshop on Computer Architecture for Machine Perception, CAMP'93, New Orleans, USA, pp. 400-409, January 1993.
- [15] C. Samson, B. Espiau, M. Le Borgne : *Robot Redundancy : an Automatic Control Approach*, NATO Advanced Research Workshop on Robots with Redundancy, Salo, Italia, June 1988.
- [16] C. Samson, M. Le Borgne, B. Espiau : *Robot Control : The Task Function Approach*, Oxford University Press, 1991.
- [17] A.C. Sanderson, L. Weiss : *Adaptive Visual Servo Control of Robots*, In Pugh, A. (editor), Robot Vision, I.F.S. Publications Ltd., pp. 107-116, 1983.
- [18] J.P. Urban, G. Motyl, J. Gallice : *Real-time Visual Servoing using Controlled Illumination*, International Journal of Robotics Research, Vol 13, n.1, pp. 93-100, February 1994.
- [19] S. Venkatesan, C. Archibald : *Real-time Tracking in Five Degrees of Freedom Using Two Wrist-mounted Laser Range Finders*, IEEE Int. Conference Robotics and Automation, pp. 2004-2010, 1990.
- [20] M.R. Ward, L. Rossol, S.W. Holland, R. Dewar : *CONSIGT : A Practical Vision Based Robot Guidance System*, Proc of 9<sup>th</sup> Int. Symp. on Industrial Robotics, pp. 195-211, Washington, U.S.A, March 1979.
- [21] L.E. Weiss, A.C. Sanderson, C.P. Neuman : *Dynamic Sensor-Based Control of Robots with Visual Feedback*, IEEE Journal of Robotics and Automation, Vol. RA-3, n. 5, pp. 404-417, October 1987.

